

Three Functions of the shortest distance

Write three functions **distance1**, **distance2**, and **distance3**, which works according to the provided comments.

```
def distance1(x1, y1, x2, y2):
    # returns the distance between points (x1, y1) and (x2, y2)
    # Usage example: d1 = distance1(0.0, 0, 3, 4) -> d1 = 5.0

def distance2(p1, p2):
    # p1 and p2 are lists
    # each list is a point, which has 2 indices, storing x and y
    # returns the distance between p1 and p2
    # Usage example: d2 = distance2([0.0, 0], [3, 4]) -> d2 = 5.0

def distance3(c1, c2):
    # c1 and c2 are lists that represent circles
    # each list has 3 indices, storing center x and y, and radius
    # returns the distance between the center of c1 and c2, as well
    # as display if the circle c1 and c2 are overlapping or not
    # Usage example: d3, overlap = distance3([0.0, 0, 1], [5, 0, 2])
    # -> d3 = 5.0, overlap = False

def perimeter(points):
    # points is a list of points
    #     each point is a list with 2 indices (storing x and y)
    #     these points are the corners of the polygon (for k-gon,
    #     there are k points total, k>=3)
    # returns the perimeter of the polygon that is defined by the
    # input points

exec(input().strip()) #must have this line when submitting to grader
```

Input

Python code for the functions

Output

The results after executing said codes.

Example

Input (from keyboard)	Output (on screen)
<code>print(distance1(0, 0, 3, 4))</code>	<code>5.0</code>
<code>print(distance2([0,0], [3,4]))</code>	<code>5.0</code>
<code>a,b = distance3([0,0,1], [5,0,2]);print(a, b)</code>	<code>5.0 False</code>
<code>print(perimeter([[0,0], [0,2], [2,2], [2,0]]))</code>	<code>8.0</code>